Your Best Buys in Hardware and Software

COMPUTER SHOPPER®

JANUARY 1987
VOL. 7 NO. 1
ISSUE 84
$2.95
© 02927
Canada $3.50

New Money Saving Tips • Manufacturer Up-Dates • In-Depth Articles

HYUNDAI –

Bets its 'Blue Chip'

What's New in 1987 Computers & Peripherals

80386 Computers Star at Fall COMDEX

Market-Trend Analysis Program

BEST PRICES - BEST PRODUCTS - BEST SERVICE
MidWest Micro-Peripherals
1-800-423-8215
12 FULL PAGES OF SAVINGS
SEE AD INDEX

DISK LIQUIDATION!
SEE OUR FULL PAGE AD FOR DETAILS
NATIONAL LIQUIDATORS
(800) 621-8272
29¢
DS/DD WITH SLEEVES

70% OFF! Most Items BCE
LOOK FOR THE RED PAGES

American Semiconductor
Computers, Components, Hardware and Software
TURBO XT $399!
256K...$1.99!
1-800-237-5758

44MB 29MS DRIVE $775.00 DEALER
Proto PC, Inc. 612-644-4660

PRINTERS Plus
800-562-2727

America's Mailorder Headquarters
Lyco Computer
World's Leader in Sales and Service
See Pages 10 & 11

N.D.R.C.
Quality Product
Discount Prices
2400 Baud Modem $249.00
40Mb Kit $549.00
(214) 750-9889

COMMUNICATIONS ELECTRONICS INC.™
25¢ Diskettes
800-USA-DISK
See ad index

Have *Computer Shopper* delivered to your home.
Call Toll Free 1-800-327-9926.
In Florida call 1-305-269-3211.

See Advertiser's Index Page 4

ATARI

Applying The Atari

by Jeff Brenner

Wow...1987 is upon us already, and the 8-bit Atarians are clamoring louder than ever! Let me hear from you, folks, with your ideas for "Applying The Atari" in 1987. Anything goes—applications software, hardware projects, reviews, etc. We've had some terrific programs listed here in 1986, as is evident from readers' responses, and 1987 promises to be equally exciting. This month we'll enter the final part to November's feature program, Forms Generator and we'll also answer some reader mail.

Reader Mail

Dear Jeff:

I have an Atari 800XL and have recently become interested in assembly language programming. I am presently working on a project which involves changing some routines in Atari BASIC. For this reason, I read with interest your column concerning copying the ROM BASIC to disk ("Atari Help," April 1986). Could you print the assembly listing of the program? Also, I would like to add line numbering to Atari BASIC and would appreciate if you could give me some advice about how and where to start this routine.

Gary Harren
Anniston, Alabama

You'll find the assembly listing under the "Disk-Based BASIC" heading. This routine runs as an AUTORUN.SYS file and loads the BASIC language from the file D:BASIC. The routine also contains a special SYSTEM RESET handler. Normally, whenever the RESET key is pressed, a GRAPHICS 0 screen is set up at the top of free RAM. When the BASIC in ROM is being used (internal BASIC on XL/XE or BASIC cartridge), the screen is set up directly below the BASIC ROM, at the top of RAM (in systems with at least 40K memory). But when BASIC is in RAM, as it is when it is loaded from the disk, this screen is set up right on top of a portion of the BASIC language, and causes a system crash. The SYSTEM RESET handler eliminates this problem by automatically replacing 1K of the damaged BASIC code after RESET is pressed.

An additional activity of this routine is to trick DOS into believing a cartridge is present. Normally, when you call up DOS and choose the "RUN CARTRIDGE" option (to return to BASIC), DOS checks if a cartridge is present and will give a "NO CARTRIDGE" message if it doesn't find one. The Disk-Based BASIC routine changes specific memory locations in either DOS 2 or DOS 2.5 whenever DOS is called to make the DOS believe that a cartridge is actually present. This way, the "RUN CARTRIDGE" function returns you to the RAM-based BASIC.

I've added many comments to the assembly listing so you, Gary, and other readers interested in assembly language can see precisely how the above-described functions are implemented. Additionally, I've listed another routine for your interest, under the "BASIC TO STRING CONVERTER" heading. This short routine is used to transfer the BASIC ROM into a string (i.e. A = USR(ADR(1536),ADR($)). The string can then be saved to the disk as the "D:BASIC" file. April's Disk-Based BASIC program used this short routine for exactly this purpose.

Regarding your question about creating an autonumbering program for the Atari, the upper half of page six (location 1792 or $680) is a safe place for the routine (or the entire page six if you need more than 128 bytes). I don't know if you are familiar yet with interrupt vectors, but you would probably want to install your routine into the keyboard handler, or the vertical blank routine. This way you can check when RETURN is pressed or when the keyboard register (764) is cleared (containing a 255 after having contained the internal code for the RETURN key, 12), and then display the next line number. A number of autonumbering routines have already been created and you can probably find one in the public domain, should you decide not to embark on this programming task yourself.

Dear Jeff:

Is there a program somewhere for word processing that uses SCRIPT type of letters? My wife would like to use something like that for some of her correspondence and I said I'd find out. I am enclosing a clipping that has the typestyle I'm thinking of. It wouldn't have to be that exact style but something like it.

Along with my Atari 800XL, I use an Epson Spectrum LX-80 printer. The Epson, as versatile as it is, should be able to do the work if there is such a program available. Many thanks for all the good work and I love to read your column.

Darwin K. Garrison
Olathe, Kansas

I don't know of any Atari word processor that allows the printing of script type (unless, of course, the printer itself supports this typestyle). However, there are still a couple of ways in which you can send out correspondence in cursive print. First of all, you'll need a word processor, such as AtariWriter Plus, that can "print" to a disk file. The original AtariWriter cannot be used for this purpose since it does not support print-

continued on page 112

DISK-BASED BASIC Assembly Listing

```
10 ; BASIC LOADER
20 ; COPYRIGHT 1986 JEFF BRENNER
30 ; Supports SYSTEM RESET, 130XE RAMDISK, and the "RUN CARTRIDGE"
40 ; option on DOS 2 and DOS 2.5.
50 ; When saved as AUTORUN.SYS, will load and execute cartridge file
60 ; saved as "D:BASIC"
70 ; 48K minimum RAM required.
80 ; Page six can be used (if SYSTEM RESET not pressed and RUN CARTRIDGE
90 ; DOS option not used) by resetting $222 and $223.  Original values
0100 ; stored in $6FE and $6FF.  To do this in BASIC:
0110 ; POKE 54286,0:? :POKE 546,PEEK(1792):POKE 547,PEEK(1793):POKE 54286,64
0120  *=$5D7 ; Locations $5D7 to $5FF are used only temporarily for
0130 ; the powerup routine.
0140 MEMLO=$2E7
0150 POWERUP LDA $0C ; This section executed when run for first time
0160  STA INIT+1
0170  LDA $0D
0180  STA INIT+2
0190  LDA #INIT&255
0200  STA $0C       ; Point DOSINI vector to our INIT routine.
0210  LDA #INIT/256 ; INIT will be executed after every SYSTEM RESET.
0220  STA $0D
0230 DOS25 LDX #0
0240  STX 580
0250  INX
0260  STX 9
0270  LDA #160
0280  STA 106      ; Reserve 8K RAM
0290  JSR SCREEN  ; Setup GRAPHICS 0 screen
0300  LSR 709  ; Blank screen while loading
0310  LDY #3
0320  STY $342+$30
0330  INY
0340  STY $34A+$30
0350  LDA #FILE&255  ; FILE holds "D:BASIC" file name
0360  STA $344+$30
0370  LDA #FILE/256
0380  STA $345+$30
0390  LDX #48
0400  JSR $E456      ; Open file
0410  LDA #7
0420  STA $342+$30
0430  LDA #250
0440  STA $344+$30
0450  LDA #159
0460  STA $345+$30
0470  DEC $349+$30
0480  JSR $E456      ; Load BASIC
0490  ASL 709        ; Restore screen
0500  LDA #$77
0510  STA 764        ; Force SHIFT-INSERT to clear last screen line
0520  LDA MEMLO
0530  STA $02
0540  STA HOLD
0550  LDA MEMLO+1
0560  STA $03
0570  STA HOLD+1
0580  CLC
0590  ADC #4
0600  STA MEMLO+1    ; Move low memory pointer up 1K
0610  LDA #DOSJUMP&255  ; Set RUN CARTRIDGE address
0620  STA $BFFA
0630  LDA #DOSJUMP/256
0640  STA $BFFB
0650 CONT LDX #4 ; Temporarily save 1K of BASIC below new MEMLO
0660  LDY #0
0670 LOOP LDA $BC00,Y
0680  STA ($02),Y
0690  INY
0700  BNE LOOP
0710  INC $03
0720  INC LOOP+2     ; Increment high byte of $BC00 at LOOP+2
0730  DEX
0740  BNE LOOP
0750  LDA #$BC
0760  STA LOOP+2  ; Restore original value at LOOP+2
0770  BNE RETURN ; Go to run BASIC
0780 ; The following section is executed each time SYSTEM RESET is pressed:
0790 INIT JSR $0000   ;$0000 is replaced with DOSINI address
0800  LDA #160
0810  STA 106   ; Set RAMTOP back again to reserve 8K
0820  JSR SCREEN ; Open GRAPHICS 0 screen
0830  JSR RESET  ; Restore MEMLO to original value
0840  CLC
0850  ADC #4
0860  STA MEMLO+1 ; Reserve 1K again
0870  LDX #4
0880  LDY #0
0890 LOOP2 LDA ($02),Y ; Restore last 1024 bytes of BASIC
0900 ADR STA $BC00,Y
0910  INY
0920  BNE LOOP2
0930  INC $03
0940  INC ADR+2 ; Increment high byte of $BC00 above
0950  DEX
0960  BNE LOOP2
0970  LDA #$BC
0980  STA ADR+2  ; Restore original value at ADR+2
0990 RETURN LDA $222
1000  STA VB+1        ; Save original vblank routine address
1010  LDA $223
1020  STA VB+2
1030  LDX #NEWVB/256
1040  LDY #NEWVB&255
1050  LDA #$06
1060  JSR $E45C       ; Install NEWVB routine into vblank
1070  JMP $A000
1080 SCREEN LDA #3   ; Subroutine to open GRAPHICS 0 screen
1090  STA $342+$20
1100  LDA #SCHAR&255
1110  STA $344+$20
1120  LDA #SCHAR/256
1130  STA $345+$20
1140  LDA #12
1150  STA $34A+$20
1160  LDX #$20
1170  JMP $E456  ; RTS after $E456 ends this subroutine
1180 RESET LDA HOLD  ; Subroutine to restore original MEMLO value
1190  STA 2       ; Use locations 2 and 3 as a page zero address
1200  STA MEMLO
1210  LDA HOLD+1
1220  STA 3
1230  RTS
1240 DOSJUMP JSR RESET
1250  JMP CONT
1260 FILE .BYTE "D:BASIC",$9B ; File name of BASIC file
1270 SCHAR .BYTE "S" ; "S" used to open to screen
1280 HOLD .BYTE 0,0  ; Reserve two bytes to hold MEMLO address
1290 NEWVB LDA $26F9 ; Vblank routine (executed continuously)
1300  CMP #$FD        ; Check if DOS 2 is being used
1310  BNE NEXTDOS     ; Nope - Check for DOS 2.5
1320  DEC $26F9       ; Make DOS 2 think cartridge is present
1330  BNE VB
1340 NEXTDOS LDA $2759 ; Check if DOS 2.5 is being used
1350  CMP #$FD
1360  BNE VB          ; Nope
1370  DEC $2759       ; Make DOS 2.5 think cartridge is present
1380 VB JMP $0000     ; $0000 is replaced with original imm. vblank address
1390  .END
```

Applying The Atari

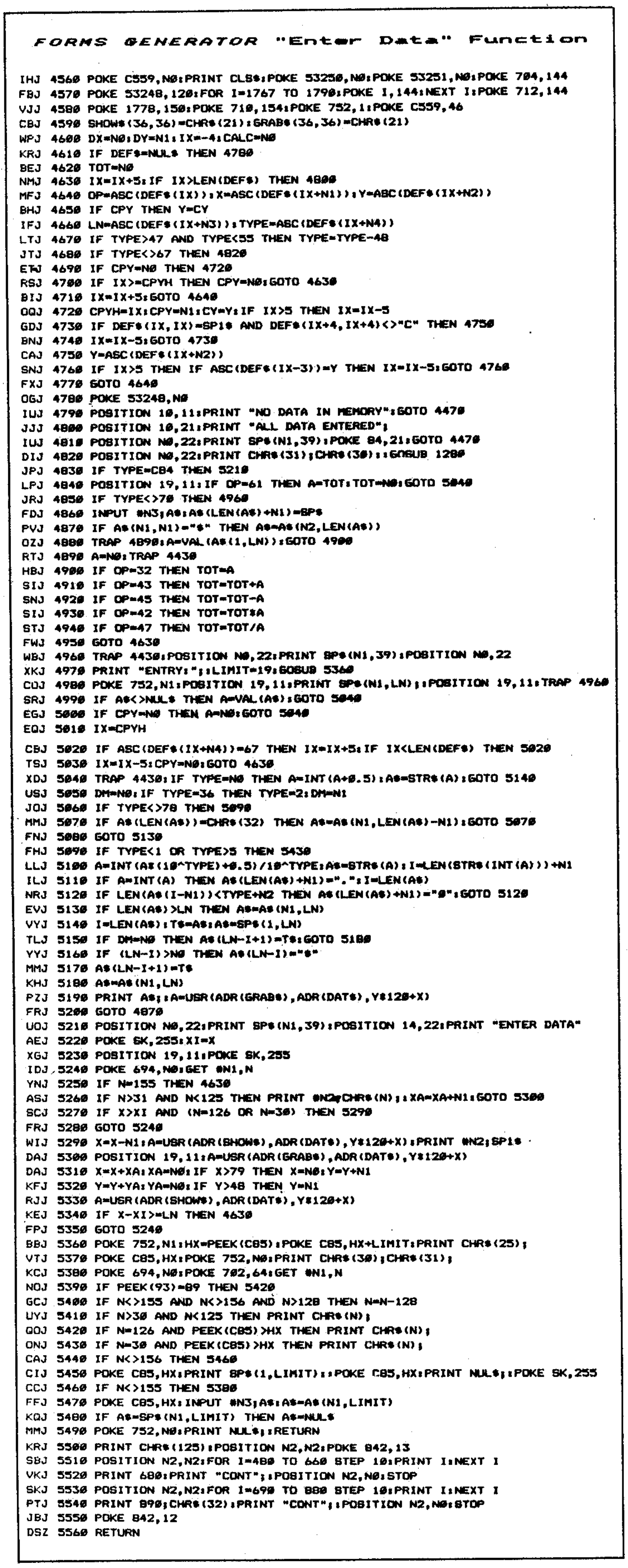
continued from page 111

ing to devices other than the printer. When you "print" the file to diskette, you are instructing the word processor to save your text in the exact way it would send the data to your printer. Therefore, your "printed" file will fit your specified margins, will have been centered where instructed to be centered, will number pages in the file, etc. Once the diskette file is in printed form, you could use XLENT Software's MegaFont to print the file with a cursive font. You could also use ATASCII Lister (published in the July 1985 "Applying The Atari), along with a character set of your own design or a public domain script character. Be forewarned, however, that with either of these programs, the characters will probably not be as readable as you would like, since these characters are defined by the same number of bytes (eight) as the standard Atari characters.

Completed Forms Generator

Load your Forms Generator program from November and make the modifications or additions shown under the "Forms Generator Changes/ Additions" heading. (**Important note:** These and the following modifications apply only to those using the listing version of the program. The standard and enhanced diskette versions have already incorporated these additions and should not be modified.)

SAVE the partially updated Forms Generator. Now load Program Perfect and use it to enter lines 4560 through 5560 under the "FORMS GENERATOR Enter Data Function" heading. These lines add the "Enter Data and Calculate" feature to Forms Generator. When these lines have been entered, LIST them temporarily to disk (i.e. LIST "D:TEMP"). Now, load the partially updated Forms Generator (with the changes/ additions incorporated) and ENTER the temporary file (i.e. ENTER "D:TEMP"). The result will be the complete standard Forms Generator program in memory, and you can now SAVE this program under a name of your choice.



FORMS GENERATOR "Enter Data" Function

```
IHJ 4560 POKE C559,N0:PRINT CLS$:POKE 53250,N0:POKE 53251,N0:POKE 704,144
FBJ 4570 POKE 53248,120:FOR I=1767 TO 1790:POKE I,144:NEXT I:POKE 712,144
VJJ 4580 POKE 1778,150:POKE 710,154:POKE 752,1:POKE C559,46
CBJ 4590 SHOW$(36,36)=CHR$(21):GRAB$(36,36)=CHR$(21)
WPJ 4600 DX=N0:DY=N1:IX=-4:CALC=N0
KRJ 4610 IF DEF$=NUL$ THEN 4780
BEJ 4620 TOT=N0
NMJ 4630 IX=IX+5:IF IX>LEN(DEF$) THEN 4800
MFJ 4640 OP=ASC(DEF$(IX)):X=ASC(DEF$(IX+N1)):Y=ASC(DEF$(IX+N2))
BHJ 4650 IF CPY THEN Y=CY
IFJ 4660 LN=ASC(DEF$(IX+N3)):TYPE=ASC(DEF$(IX+N4))
LTJ 4670 IF TYPE>47 AND TYPE<55 THEN TYPE=TYPE-48
JTJ 4680 IF TYPE<>67 THEN 4820
ETJ 4690 IF CPY=N0 THEN 4720
RSJ 4700 IF IX>=CPYH THEN CPY=N0:GOTO 4630
BIJ 4710 IX=IX+5:GOTO 4640
OQJ 4720 CPYH=IX:CPY=N1:CY=Y:IF IX>5 THEN IX=IX-5
GDJ 4730 IF DEF$(IX,IX)=SP1$ AND DEF$(IX+4,IX+4)<>"C" THEN 4750
BNJ 4740 IX=IX-5:GOTO 4730
CAJ 4750 Y=ASC(DEF$(IX+N2))
SNJ 4760 IF IX>5 THEN IF ASC(DEF$(IX-3))=Y THEN IX=IX-5:GOTO 4760
FXJ 4770 GOTO 4640
OGJ 4780 POKE 53248,N0
IUJ 4790 POSITION 10,11:PRINT "NO DATA IN MEMORY":GOTO 4470
JJJ 4800 POSITION 10,21:PRINT "ALL DATA ENTERED";
IUJ 4810 POSITION N0,22:PRINT SP$(N1,39):POKE 84,21:GOTO 4470
DIJ 4820 POSITION N0,22:PRINT CHR$(31);CHR$(30);:GOSUB 1280
JPJ 4830 IF TYPE=C84 THEN 5210
LPJ 4840 POSITION 19,11:IF OP=61 THEN A=TOT:TOT=N0:GOTO 5040
JRJ 4850 IF TYPE<>70 THEN 4960
FDJ 4860 INPUT #N3;A$:A$(LEN(A$)+N1)=SP$
PVJ 4870 IF A$(N1,N1)="$" THEN A$=A$(N2,LEN(A$))
OZJ 4880 TRAP 4890:A=VAL(A$(1,LN)):GOTO 4900
RTJ 4890 A=N0:TRAP 4430
HBJ 4900 IF OP=32 THEN TOT=A
SIJ 4910 IF OP=43 THEN TOT=TOT+A
SNJ 4920 IF OP=45 THEN TOT=TOT-A
SIJ 4930 IF OP=42 THEN TOT=TOT*A
STJ 4940 IF OP=47 THEN TOT=TOT/A
FWJ 4950 GOTO 4630
WBJ 4960 TRAP 4430:POSITION N0,22:PRINT SP$(N1,39):POSITION N0,22
XKJ 4970 PRINT "ENTRY:";:LIMIT=19:GOSUB 5360
COJ 4980 POKE 752,N1:POSITION 19,11:PRINT SP$(N1,LN);:POSITION 19,11:TRAP 4960
SRJ 4990 IF A$<>NUL$ THEN A=VAL(A$):GOTO 5040
EGJ 5000 IF CPY=N0 THEN A=N0:GOTO 5040
EQJ 5010 IX=CPYH
CBJ 5020 IF ASC(DEF$(IX+N4))=67 THEN IX=IX+5:IF IX<LEN(DEF$) THEN 5020
TSJ 5030 IX=IX-5:CPY=N0:GOTO 4630
XDJ 5040 TRAP 4430:IF TYPE=N0 THEN A=INT(A+0.5):A$=STR$(A):GOTO 5140
USJ 5050 DM=N0:IF TYPE=36 THEN TYPE=2:DM=N1
JOJ 5060 IF TYPE<>70 THEN 5090
MMJ 5070 IF A$(LEN(A$))=CHR$(32) THEN A$=A$(N1,LEN(A$)-N1):GOTO 5070
FNJ 5080 GOTO 5130
FHJ 5090 IF TYPE<1 OR TYPE>5 THEN 5430
LLJ 5100 A=INT(A*(10^TYPE)+0.5)/10^TYPE:A$=STR$(A):I=LEN(STR$(INT(A)))+N1
ILJ 5110 IF A=INT(A) THEN A$(LEN(A$)+N1)=".":I=LEN(A$)
NRJ 5120 IF LEN(A$(I-N1))<TYPE+N2 THEN A$(LEN(A$)+N1)="0":GOTO 5120
EVJ 5130 IF LEN(A$)>LN THEN A$=A$(N1,LN)
VYJ 5140 I=LEN(A$):T$=A$:A$=SP$(1,LN)
TLJ 5150 IF DM=N0 THEN A$(LN-I+1)=T$:GOTO 5180
YYJ 5160 IF (LN-I)>N0 THEN A$(LN-I)="$"
MMJ 5170 A$(LN-I+1)=T$
KHJ 5180 A$=A$(N1,LN)
PZJ 5190 PRINT A$;:A=USR(ADR(GRAB$),ADR(DAT$),Y*120+X)
FRJ 5200 GOTO 4870
UOJ 5210 POSITION N0,22:PRINT SP$(N1,39):POSITION 14,22:PRINT "ENTER DATA"
AEJ 5220 POKE SK,255:XI=X
XGJ 5230 POSITION 19,11:POKE SK,255
IDJ 5240 POKE 694,N0:GET #N1,N
YNJ 5250 IF N=155 THEN 4630
ASJ 5260 IF N>31 AND N<125 THEN PRINT #N2;CHR$(N);:XA=XA+N1:GOTO 5300
SCJ 5270 IF X>XI AND (N=126 OR N=30) THEN 5290
FRJ 5280 GOTO 5240
WIJ 5290 X=X-N1:A=USR(ADR(SHOW$),ADR(DAT$),Y*120+X):PRINT #N2;SP1$
DAJ 5300 POSITION 19,11:A=USR(ADR(GRAB$),ADR(DAT$),Y*120+X)
DAJ 5310 X=X+XA:XA=N0:IF X>79 THEN X=N0:Y=Y+N1
KFJ 5320 Y=Y+YA:YA=N0:IF Y>48 THEN Y=N1
RJJ 5330 A=USR(ADR(SHOW$),ADR(DAT$),Y*120+X)
KEJ 5340 IF X-XI>=LN THEN 4630
FPJ 5350 GOTO 5240
BBJ 5360 POKE 752,N1:HX=PEEK(C85):POKE C85,HX+LIMIT:PRINT CHR$(25);
VTJ 5370 POKE C85,HX:POKE 752,N0:PRINT CHR$(30);CHR$(31);
KCJ 5380 POKE 694,N0:POKE 702,64:GET #N1,N
NOJ 5390 IF PEEK(93)=89 THEN 5420
GCJ 5400 IF N<>155 AND N<>156 AND N>128 THEN N=N-128
UYJ 5410 IF N>30 AND N<125 THEN PRINT CHR$(N);
QOJ 5420 IF N=126 AND PEEK(C85)>HX THEN PRINT CHR$(N);
ONJ 5430 IF N=30 AND PEEK(C85)>HX THEN PRINT CHR$(N);
CAJ 5440 IF N<>156 THEN 5460
CIJ 5450 POKE C85,HX:PRINT SP$(1,LIMIT);:POKE C85,HX:PRINT NUL$;:POKE SK,255
CCJ 5460 IF N<>155 THEN 5380
FFJ 5470 POKE C85,HX:INPUT #N3;A$:A$=A$(N1,LIMIT)
KQJ 5480 IF A$=SP$(N1,LIMIT) THEN A$=NUL$
MMJ 5490 POKE 752,N0:PRINT NUL$;:RETURN
KRJ 5500 PRINT CHR$(125):POSITION N2,N2:POKE 842,13
SBJ 5510 POSITION N2,N2:FOR I=480 TO 660 STEP 10:PRINT I:NEXT I
VKJ 5520 PRINT 680:PRINT "CONT";:POSITION N2,N0:STOP
SKJ 5530 POSITION N2,N2:FOR I=690 TO 880 STEP 10:PRINT I:NEXT I
PTJ 5540 PRINT 890;CHR$(32):PRINT "CONT";:POSITION N2,N0:STOP
JBJ 5550 POKE 842,12
DSZ 5560 RETURN
```

Program Changes

After you've entered the additions and modifications (lines 25 to 3760), you will notice a difference when Forms Generator is run: Before the "PLEASE STAND BY" message appears, a column of numbers is displayed, twice, and the cursor will be seen moving from the top to the bottom of the screen. Actually, what is happening is that the program is deleting some of the initialization lines of the program. This frees up enough memory for the rest of the program, along with the added "ENTER DATA" function, to operate. Once the lines have been deleted, the program displays the "PLEASE STAND BY" and then operates as normal. However there is one important note you should be aware of: Once you have run the updated Forms Generator, do not SAVE it. Since many lines of the program essential to the initialization procedure are deleted to free up more memory, you will be saving an incomplete program. Therefore, if you discover you need to make additional changes to the program after you run it, first load in the complete program, make the changes, and save the program before running it.

Another change made in the program is the replacement of the CONSTANT definition with the FORMS definition (used in the DEFINE CALCULATIONS mode). Essentially, the two definitions are equivalent; the name was changed to more accurately reflect the function of the definition. When you are defining calculation and want to incorporate a constant that appears on the form, you would now type F instead of C for the "DECIMAL PLACES, N, $..." prompt. Hence, if you want to multiply a subtotal on a form by a tax percentage, you would define the tax amount (such as .06) as a FORMS variable—signifying to the program that the value is taken from the form itself and not from user input.

The more important reason for the name change, however, is that values that Forms Generator itself places on the screen can be used in future calculations. For example, let's say you define the first line of your form as quantity * unit price = extended price. Then you copy this definition for the next 30 lines of your form (by pressing "C" for the "COMMAND OR RETURN TO MARK FIELD BEGIN" prompt). When you later enter the quantity and unit price information for each line, Forms Generator will automatically calculate the extended price and place it on the form. Hence, you can tell Forms Generator to sum up the extended price column to give you a subtotal or total. To do this, you would go back to each extended price box and define it now as a FORMS variable, while adding them together. This tells Forms Generator to go back and take the value that it put in the extended price box and use that in an additional calculation (calculating the total).

One other feature added: When you have a form with a lot of lines of "copied" definitions, but have finished entering your data, simply press RETURN to the ENTRY prompt and Forms Generator will skip the remaining copied lines. For example, if you copied thirty lines of the quantity * unit price = extended price formula, but only have four lines of information to enter, you would type in the data for those four lines and then press RETURN by itself. Forms Generator will then bypass the remaining copied lines and will proceed with the next definitions (such as calculating a subtotal).

Using "Enter Data" Function

This function enables you to actually enter the data according to the calculation definitions you defined. For numerical definitions, you'll see an "ENTRY" line on the bottom of the screen. Type in your entry and press RETURN. The value will be reproduced on the form in its proper format. As you enter the necessary data, the defined calculations will be performed and the results will appear on the form. For text definitions, the cursor will be positioned on the form itself, allowing you to type the text directly onto the form at the specified position.

Notes on Defining Calculations

Defining calculations for a form can get quite complicated, and unless you are experienced with defining formulas (in programs such as VisiCalc, SynCalc, etc.) you will probably want to practice and experiment to best learn how to properly and efficiently define calculations for your forms.

And a note for more experienced users: If you are creating a complicated form, with numerous calculation-definitions, it is wise to save your form (definitions are automatically saved with the form) after every major string of operations. This way, in the event that you accidentally enter an incorrect formula, or make any mistake, you can always reload the form and begin where you last saved the definitions (by selecting ADD after choosing the DEFINE CALCULATIONS option).

Next Month

We'll give the completed Forms Generator a practical test by entering a programmed form. Another form will be included that takes advantage of the graphics capabilities of the enhanced diskette-version of Forms Generator. Plus, we have a special hardware project for you loyal 800 owners, thanks to Roger E. King of SPACE (Harrisburg, Pennsylvania). (More reader mail too!)

Readers' questions, comments and contributions are welcome. Please enclose a self-addressed, stamped envelope (SASE) for a personal reply. Due to volume of letters received, only a selected number of personal responses can be given each month.

The Forms Generator program disk, from the November 1986 issue is available for *Computer Shopper* readers from the author for $7.00 for the standard version, or $11.00 for the enhanced version with documentation. Please make checks payable to "Jeff Brenner" and specify your disk drive model.

"Program Perfect" is a utility used to check for typing errors while entering programs from this column. Readers can send $5 for a diskette or a SASE for a listing of this program.

Address all correspondence to: Jeff Brenner, "Applying The Atari 1/87", c/o *Computer Shopper*, P.O. Box F, Titusvillle, FL 32781-9990. ●

ATARI ST

Hacking the ST

by Wynn Rostek

Last month we showed the Setscreen XBIOS call. One of the problems with the Setscreen call was that the operating system always performed a clear screen after each Setscreen call. I promised to go over switching the screen display area this month, so here goes.

To actually switch the screen display area, we need to look at the way the ST goes about generating a display. As you know if you have been following this column, the ST uses a bit-mapped display.

For those of you who may not have heard the term before, a bit-mapped display is one where each pixel on the screen is actually one or more bits of memory. In the old days, the memory used for holding the text displayed on a terminal was very small. The characters that were sent to the terminal were stored in a small buffer that was only about 2K! Keep in mind, these were the days when memory was very expensive.

Each character received was stored in one of the bytes of memory, in ASCII form. When the time came to display a pixel from one of the characters in memory, the ASCII value formed part of the address that was supplied to a ROM. The other part of the address was the scan line number, or rather the lowest 3 bits of the scan line number.

With a 5 by 7 matrix, the ASCII character in memory location 0 of our hypothetical terminal has to generate 7 lines, each 5 bits long. The terminal actually generates 8 lines of 6 bits, to allow some space between characters.

On scan line 0, the ASCII character in memory location is input as an address along with the three low bits of the scan line number. The ROM outputs 6 bits, and each is sent to the CRT. The terminal then uses the next 63 bytes of memory in the same manner. We are then back to using the ASCII character in memory location 0 and scan line 1 as the input to the ROM.

Figure 1 shows a way of dividing up the address lines so that we can generate a proper display. The important thing to note is that the ASCII character stored in memory location 0 is used 8 times to generate the display. This allows us to show 16 lines of 64 characters in 8 by 6 pixels using less than 2K! This is a total of 49152 pixels generated from less than 16384 bits of RAM.

While using a ROM to store the actual bit pattern of the characters helps save on RAM, we are limited in the characters we can display. In a bit-mapped display we have one bit of memory for each pixel. When a bit of memory is a 1, a dot appears on the screen.

This means that instead of decoding the character each time a pixel is needed, we decode the character once, and set the bits in memory as needed. The disadvantage of this approach is that we need more RAM for the same number of displayed characters but now since we are using a single bit of memory per pixel, we can draw lines!

Since RAM has become so cheap, most all systems today use the bit-mapped display technology. With a bit-mapped display there are two important memory addresses the system needs. The first address is the address where the bit map starts in memory, and we also need an address where the bits are being turned on when something is written to the screen.

In most systems, the address of the display buffer (the bit map) is fixed by the system. The IBM PC and most clones use this approach. In the ST, the bit map can be anywhere in memory.

To set the starting address of the bit map, two bytes are written to hardware registers within the ST. The address used for the bit map is actually 24 bits long, but the ST assumes that the bit map occurs on page boundary. If the address occurs on a page boundary, it is a multiple of 256. Any multiple of 256 has the lower 8 bits of the value as 00 hex, so the extraneous 00 is not stored (saving a register in expensive custom chips).

The second and third bytes of the address are not 0, and these must be written to a hardware register for the system to display the bit map.

Unlike many systems, the ST can display one bit-map, while writing to another. The two hardware registers are used to designate the address of the bit map that the ST displays, but a memory location is used to specify the bit map area that is to be written to.

Listing 1 (pg. 242) is a program to show the ST using two different bit maps. Notice that we switch the bit map by performing pokes to different addresses. The high addresses (ffffxxx hex) are really the hardware register addresses in the ST. Listing 2 (pg. 242) shows some assembly language routine used in this program. The peek and poke functions shown here are run in the supervisor mode so you avoid the infamous "2 Bombs" syndrome.

continued on page 242

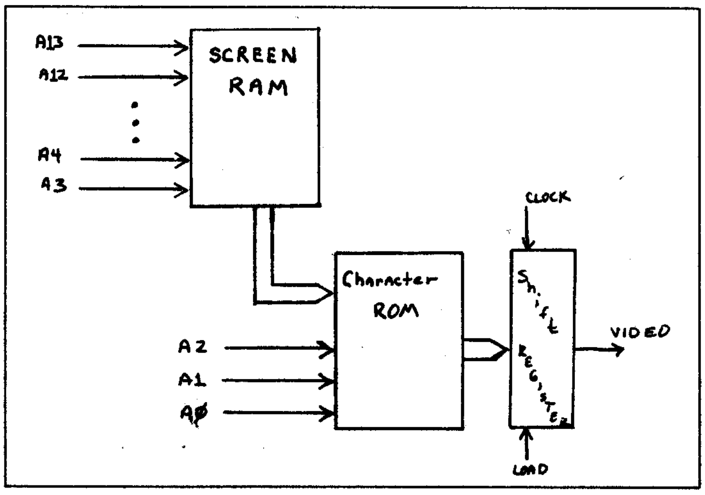
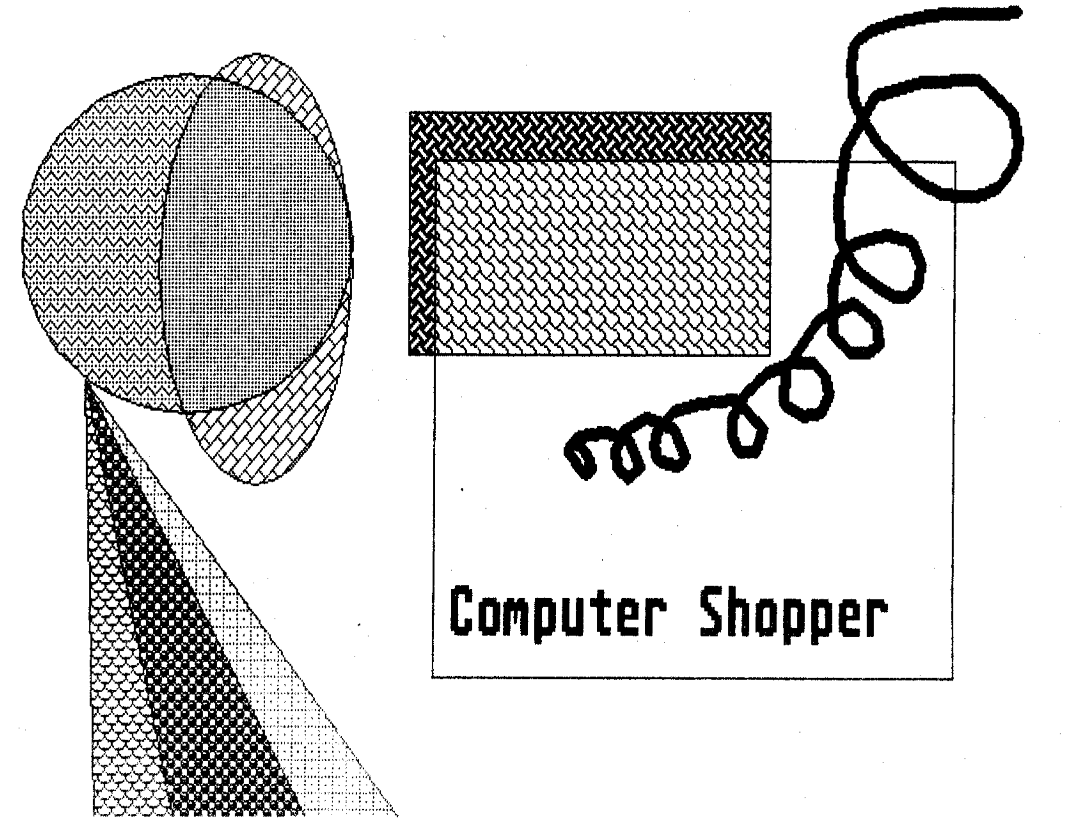


Figure 1—Basic Video Display

Degas Elite From Batteries Included



by John J. Anderson

There are a few disks which are software essentials in any Atari ST library. One is a word processor, another a telecommunications program, yet another is a paint program of some kind. After all, what's a computer like the ST for if you can't get some eye-popping graphics on the screen?

When it comes to paint programs for the ST, although there are a number of choices out there, the original Degas package from Batteries Included quickly emerged as the program of choice. It was truly impressive in its graphics capabilities, yet was easy-to-learn and easy-to-use. It was able to work in both color modes as well as monochrome mode, and offered features that outstripped every other package available. Without Degas, no ST software library could truly be called complete.

And yet, there were a number of quiet omissions in Degas, which limited its capabilities quite a bit. Four omissions could be called critical: There was no zoom or magnification mode, no scaling potential, no ability to rotate blocks, and no space for primitives to reside. Even lacking these, the program was still very good. But if these important features could somehow be incorporated...

After a wait that seemed at times interminable, Batteries Included has finally brought us Degas Elite. It includes all the features listed above, plus a whole lot more. It is more

continued on page 247

CP/M For The Atari ST

by W. Krieger

As I write this it's the end of October, and I've just received what I consider a Christmas present from Atari Corporation. The long awaited CP/M emulator is in my hands and it didn't cost me a dime. And if you want one you can get it for free as well.

Atari Corporation has just released their CP/M-Z80 emulator package into the public domain. And they uploaded it to both CompuServe and to GEnie. It may be on other services as well, but I know for sure it's up on these.

The file is CPM.ARC and is an archive file. So before you can start playing with it you will first need to "DE-ARC" it by using ARC.TTP. Which is available on both GEnie and CompuServe as well.

Of course the day it showed up on the commercial services it also started to show up on private BBSs. I know of three in Orlando, FL. that had it up the same day it was posted on GEnie. So if you are interested in this system check your local BBS before getting on the pay systems.

If it's not on your local BBS after you download it from the pay service of your choice, upload it to a local system. The pay services are great, but local BBSs are the backbone of public domain software and they should be supported by everybody who uses them.

What the CP/M-Z80 emulator is, is a software package that allows your Atari ST to run programs written for the CP/M systems, which use either an 8080 or a Z80 CPU.

You can think of it as a Z80 machine language interpreter running on your ST's 68000. It's of course slower than a real Z80 because every instruction needs to be translated to 68000 machine code. Only the speed of the 8Mhz 68000 of the ST makes this practical.

Once the 68000 can run Z80 code it's a simple matter to implement CP/M. CP/M is just another program that was written for the 8080 Z80 family of microprocessors.

Prior to the introduction of the IBM-PC, CP/M was the major disk operating system available for microcomputers. There were and still are thousands of CP/M systems in use. And of course to support these systems there is an impressive software library available for CP/M systems.

The CP/M public domain is amazing. I think that out of all the popular operating systems that have larger public domain libraries, MSDOS, Apple II, Atari 8-bit, etc., that the CP/M public domain has the widest selection, and the best quality.

One of the reasons for this is that CP/M was first. It's been around longer and has had more time to grow. Also in the early years of micro based systems, hackers and hobbyist were the market.

There wasn't perceived to be any real money to be made writing programs for micros, so the people who were writing programs at that time were writing them because they wanted to make their system do something it had never done before.

After they had gotten their program written they wanted to show it off and have others use it. They didn't market it, they gave it away, usually with source code! Which of course was hacked on and improved by everyone in the tight microcomputer community of that time.

Another reason for the strength of CP/M public domain software is that many companies don't see this as a viable market anymore and rather than market and to support their CP/M products they too have released some formally commercial products into the public domain.

I don't mean to imply that all CP/M programs are public domain, far from it. Many companies still consider CP/M as a good market and support it. You can still buy Turbo Pascal, Wordstar, dbase II, and many other programs for CP/M systems.

CP/M is far from dead, and thanks to Atari's CP/M-Z80 emulator ST, owners can now tap into the potential of this mature and viable operating system.

The first thing you need to do to use the emulator is to make some CP/M format disks. TOS and CP/M use completely different directory formats so TOS can't read CP/M disks and vice versa.

Included in the emulator package is a program MAKECOM.TOS this is used to make what the documentation refers to as a pseudo CP/M disk. It will be used to make real CP/M disks. This disk will also have two special programs on in INIDSK.COM and TOSCPM.COM.

Once you have made your first disk under TOS the rest of the process is run from under CP/M. Just run the emulator program and when prompted to insert your pseudo CP/M disk hit return.

You are now running a CP/M system. The only difference I've seen between the emulator and my "real" CP/M system is that instead of getting A> or B> as a prompt the emulator uses A: or B:.

Now is when INIDSK and TOSCPM come into play. IN-

continued on page 246

Atari ST vs. Amiga Round 2 The ST Is Superior

by D.F. Scott

There's been a lot of response to Sheldon Leemon's article in *Computer Shopper's* November issue called "The Myth of ST Superiority," wherein the author sets out to prove the arguments for the ST when compared against the Amiga are, for the most part, unfounded. The article merits strong response, because Leemon is one of the most respected and experienced technical writers in the profession. I read the article closely with my acclaimed photon stereo optiscopes, and took some notes, which I'd like to expound upon here.

My overall impression as a writer in the same profession: One cannot dispute any of Leemon's facts, it's his reasoning that I question. Throughout the article, he indulges in a personal diatribe not against the ST, but against the Tramiels. He bases his Amiga facts on the textbook and his ST facts upon what Leonard has said or Sam has said. He jumps upon their occasional exaggerations of the facts as if the machine they sell were, as a result, somehow at fault. Leemon, when pressed to find selling points for the Amiga, naturally turns to the subject of graphics at every opportunity—one strong point of a machine otherwise surrounded by weak points. But perhaps most importantly, at no time does Leemon prove that any of the five statements he claims are "myths" are unfounded or wrong! Instead, he took the facts and colored them to suit his deadline.

Specific instances: "Myth 1," for instance, that the ST is faster than the Amiga. Here, as almost everywhere else, Leemon jumps onto the subject of graphics; while barely mentioning the fact that it is the Amiga which has the "bimmer" chip that's meant to speed it up. Even with the bimmer, Leemon himself concedes the ST is faster by "at best 15%." He does correctly state the Amiga's custom co-processor chips, although stealing CPU time only occasionally, can plot graphics while the CPU is otherwise occupied; whereas the ST's CPU handles the graphics—Leemon exaggerates here to make it seem plotting each individual point is trudgery—and that nothing else can happen "simultaneously." Still, this does not explain nor does it take away from the fact that the ST is the faster machine, and will be faster still with its coming blitter enhancement. Before a machine can have a ritzy display, it must first add and subtract—something which the Amiga has been found to do slower than the ST.

"Myth 2," that the ST's display is clearer than the Amiga's, is supported in Leemon's mind by the fact that one can buy a better monitor, a TV, or even a big-screen projection system that will show off the versatility and clarity of the Amiga display—in other words, if you don't like the Amiga display, go buy a Sony. With the ST's super-clear RGB, who wants to buy another monitor or use their bleeding TV set?

"Myth 3" is that the Amiga's OS is bug-ridden, to the extent that the machine is totally unprogrammable. Of course that would be a myth; somebody has to program the darned thing! Leemon's contention is that ST GEM being bug-ridden in places somehow evens the score, thus neutralizing all of the Amiga's notorious OS bugs. Hmmm.

Yes, we all know the Amiga "Intuition" OS has the multitasking capability the ST doesn't have. I've known some programmers here who'd laugh at that term "multi-tasking" in relation to the Amiga (and the ST, for that matter) because of how slow the multitasked programs would run. Leemon claims, though, that he can "run a word processor, format disks, and play a game simultaneously while on-line." He goes on to say, "Just think, never to be bored by a slow online conference again." Now you can guess what he's up to if he ignores you on-line.

Frankly, the way I have my Flash disk set up, I can format disks, run the Word 400 processor, and still be online, and even if it were electronically supported I couldn't actually *operate* all three applications at once! Neither could one with the Amiga; although all three applications are running (in Leemon's case, four if you count Arctic Fox or whatever), one can—and should—input to only one application at one time. Multi-tasking is multiple output, not multiple I/O; thus it's not for general-purpose systems, but for major operations like CAD/CAM. Multitasking is only really useful when the programs operating under such an environment are actually doing something even while you're not inputting data to them.

"Myth 4" is that there's more software for the ST than the Amiga, as proven by software catalogs. Perhaps Leemon's reading the *Electronic Arts* catalog; but I'll be glad to show him some from such major distributors as Ingram and C.S.S. where ST software outnumbers Amiga by as much as six to one.

"Myth 5" is that the Amiga is a game machine. Here we see the same double-standard approach other writers have taken to defend the ST—that the label is unfair for such a powerful machine as the Amiga, but on the other hand it plays some awesome games. Both the ST and the Amiga play awesome games, as they should; and I do agree that game software has to be faster, tighter, and more complex than business software—so the better game machine is the better business machine. Still Leemon bases his facts upon what Sam Tramiel has said; which leads me to state if Leemon really wanted to prove the ST inferior, he should have taken off the cover first. He puts Sam down for building a game machine, as well as speaking of the Amiga as such; when in fact Leemon agrees with him.

I like the ST for what it can do for me. I have no doubt that others will prefer the Amiga for what it can do for them. If I say the ST is superior to the Amiga, it is based upon my personal experience with both machines, not what Leonard Tramiel or Frank Leonardi may have said to someone writing an article. If someone is to make an opposing claim to superiority, he should base his facts upon what holds water and not what floats on air. ●

CP/M For Atari ST

continued from page 114

DISK initializes blank formatted disks so they can be used by CP/M. TOSCPM is a program that copies files from a TOS formatted disk to a CP/M formatted disk.

Use TOSCPM to copy NSWP207.COM onto your pseudo disk from the distribution disk. This is a public domain program, New Sweep which makes file manipulations on CP/M systems very easy. Then use INIDSK to make a couple of CP/M disks. Use New Sweep to copy itself and TOSCPM and INIDSK to your newly made CP/M disks.

Now that you have a couple of prepared disks, use TOSCPM to copy the rest of the files from the distribution disk to your CP/M disks.

Two more programs that bear special mention are CPMTOS.COM and TOSDIR.COM. CPMTOS is the reverse of TOSCPM and lets you move files from your CP/M environment back into TOS. TOSDIR allows you to read a TOS disk directory from CP/M.

Of course you still haven't got anything really useful until you port some CP/M programs onto the system. This can be done several ways. CP/M files can be downloaded from many BBSs or commercial time-share systems.

I think however that the people who will benefit most from this emulation package are people like myself who already have a CP/M machine with a collection of software they don't want to give up just because they bought into an ST.

In my case for example I have a couple of programs on my Osborne that I really like. But after working on my ST with it's nice big 80 column color monitor it's really a pain to fire up the Osborne and work with it's tiny 52 column display. It didn't used to bother me but I guess I'm getting spoiled.

Anyway, to port files to the ST's CP/M emulator is a two step process. First you need to get them onto a normal TOS disk. Then use TOSCPM to move them onto the ST-CP/M disks.

I just got a new ST telecommunication program that makes the first part of this process a breeze. The program is CONNECT ($29.95 from SST Systems, 3456 Willis Dr., Titusville, Fl. 32796) CONNECT is a good program and I now use it as my primary telcom. package. But what makes it really outstanding for the job at hand is it's REMOTE feature.

What CONNECT allows you to do is control both sides of the file transfer process from a remote system. All I had to do was to run a Null modem cable between my Osborne and my ST. Bring up CONNECT on the ST and go into REMOTE.

Then bring up MODEM7 on the Osborne, from the Osborne I was able to tell CONNECT to change drives, receive files, send files, erase files, open or close folders, etc. It was great, instead of jumping back and forth between both systems for each file, I was able to sit at the Osborne and do it all.

For me this is a feature sent down from on high. Having three systems ST, Osborne, and PC-CLONE, I'm constantly porting files between machines. CONNECT makes my life much easier.

I've even started to use it as my private BBS. I'll leave my modem hooked up with auto answer on and leave my ST on running CONNECT in REMOTE mode. Then I can call home from work and port files from the PC at work to my ST and vice versa.

If you have or regularly use two computers you should look at this program. I know that now that I have gotten used to having it I don't want to give it up.

But back to the subject at hand, once you port files over to your ST-CP/M system how well do they work? So far I have to say they work great. I have not of course been able to try even a significant portion of the CP/M programs available. But the ones I have tried all seem to work.

There is a problem with screen control. CP/M doesn't have any standard or default terminal. Most programs that do a lot of screen manipulation have to be installed for the terminal you're using. And of course none of supplied terminal settings are for an Atari ST.

Wordstar for example looks really weird, and unless you take the time to install a new screen driver for the ST it's unusable. But in my personal opinion Wordstar started out unusable so I didn't spend a lot of time playing with it.

I had no trouble installing or running Turbo Pascal. And the same is true for MBASIC and ZORK1. All the BASIC and Pascal programs I ported over ran. Also the two public domain programs supplied on the distribution disk New Sweep and NULU worked fine.

In the documentation Atari says that the emulator is equivalent to a 2MHz CP/M system on average CP/M programs. I think that this is real close if not right on.

My Osborne is a 4MHz CPU with wait states for the slower dynamic RAM that they used. Granted the ST is slower but it's not running half the speed of the Osborne, so it seems to be running at least as fast as a 2MHz system with no wait states.

I only have minor complaints about the emulator package. The most significant complaint is that the package is not set up for single drive systems. It's not difficult to use a single drive system but it's difficult to set up on one.

A lot of this is due to everything being on one distribution disk when you first get it. The documentation makes it clear that Atari had intended to distribute it on two disks, one already formatted for CP/M with the CP/M programs already on it.

Another problem that again hurts people with only one drive is that the emulator doesn't like RAM disks. You can set up a RAM disk as a TOS disk and use TOSCPM and CPMTOS to move files from it to the CP/M disk. But I wasn't able to use the RAM disk as a CP/M disk.

Also the INIDSK program is a little flaky. I've got some blank formatted disks that TOS has no trouble with, but when I tried to run the INIDSK on them I got an error message "disk is not empty" and I couldn't use them with the emulator. Some other disks from the same manufacturer formatted on the same drive work just fine.

All and all this is a good package, with only minor problems. I don't know why Atari changed their plans and instead of marketing this package decided to release it into the public domain, but I am glad they did.

The emulator was developed by SoftDesign of Germany in conjunction with Atari, so I'm sure Atari spent some bucks to get it.

My feelings are that Atari's marketing people think that there would not be enough sales of the product to cover the cost of manufacturing and distribution. I don't know if this is right but it's my best guess.

Instead of letting it be a total waste, it seems that Atari decided to release the product as public domain. This, I think may turn out to be a smart move on their part. There are a lot of Apple II users in the world who run a Z80 coprocessor and CP/M on their Apples. Many of these people will be wanting to upgrade their systems sooner or later.

Normally I think that they would tend to go with a Macintosh, but if they can get an ST for half the price and port their existing CP/M software over, the ST becomes much more attractive. The same argument holds true for the thousands of people with other CP/M machines who would also like to upgrade to a newer more powerful machine.

I wouldn't have been at all reluctant to pay a reasonable price for this program as is. Now that Atari is willing to let me have it for free all I can say is thanks Atari! If you are even remotely interested in CP/M and have an ST, get yourself a copy of this package. ●

Spreadsheet Package For The ST-PowerPlan

Since VisiCalc, the computer has become an important planning tool. By using an electronic ledger, you can perform hundreds of calculations and "what-if" analyses quickly and reduce mounds of data into meaningful information. PowerPlan ST is such a tool for the fast-selling Atari ST computer.

As the name suggests, PowerPlan ST is a powerful analysis package featuring a large spreadsheet (65536 x 65536 cells), a built in calculator and an online notepad. And it also has integrated graphics. So data from the spreadsheet can be graphically summarized in pie, bar and line charts and displayed simultaneously.

PowerPlan ST is easy to use, since it uses the familiar GEM built into the ST. And PowerPlan ST can use multiple windows—up to seven. One window can display a part of the spreadsheet; a second window, a different part; and a third, a pie or bar chart of the data. And it works with either a monochrome or color monitor.

Software such as PowerPlan ST is just another reason that the Atari ST is one of the best buys in computers today. PowerPlan ST began shipping mid-November. The suggested retail price is $79.95.

For additional information contact Abacus Software, 2201 Kalamazoo S.E., P.O. Box 7211 Grand Rapids, MI 49510; (616) 241-5510, Telex 709 101.

Mention that you read about it in *Computer Shopper*. ●

F & W SPECIALISTS IN CAD/CAM, GRAPHICS, DESKTOP PUB

SPECIAL CAD/CAM DEAL!!!

Ikegami ultra high resolution monitor $1400.00 value. Viva Silver card scompatable with Control Systems Artist One Card. Runs well with Autocad. Value of Viva Card is $995.00. The best price we've seen is $2395.00. We have packaged this together for a special combination price of only $995.00 COMPLETE!!! This package is ready to run on IBM XT and AT and compatables.

OTHER MONITOR SPECIALS	
IBM enhanced color display (new)	$449.00
19" Ikegami color monitor (used exc. cond.	$399.00
13" Ikegami color monitor (new)	$329.00
9" Panasonic monochrome open frame	$34.50
12" Panasonic monochrome open frame	$27.50
15" Panasonic monochrome open frame	$27.50

IBM SOFTWARE SPECIALS	
IBM DOS 2.1	$39.00
IBM Displaywrite 3 version 3.1	$175.00
IBM Displaywrite 2 version 1.10	$125.00
IBM PC Network Program	$50.00
IBM PC Storyboard	$75.00
IBM Data Edition	$75.00
IBM BASIC	$24.00
IBM BASIC 3.0	$29.00

IBM AT Documentation Kit:
consists of:
guide to operations AT
Basic 3.0
power cord
diagnostics
intro book
all of the above $49.00

POWER SUPPLIES	
Videotel (shown) +5v 1.8 amps -5v 50ma, +12v .5amps. Makes a dandy bench supply.	$10.95
Raytheon open frame switching supply 5v @ 10 amps +12, -12 @ 2amps	$9.75
Panasonic 5v @ 3amps +12,−12 @ ½amp	$8.95

IBM HARDWARE SPECIALS	
5¼" double side double density floppy drives full height	$49.00
1.2mb 5¼" half height floppy (new)	$149.00
IBM Printer cables	$12.00
PCjr wireless keyboards	$59.00
Prof. Graphics controller (list $2995.00)	$1495.00
PC Translator unit (for networking)	$299.00
Network Short Distance Kit	$15.00
AT 512kb memory expansion option	$249.00

PLOTTERS, DIGITIZERS, ETC	
HP 7475A 6 color plotter	$1395.00
HP 7470 1 color plotter	$595.00
HP 7225 (GPIB)	$495.00
Summagraphics MM 1201 digitizer tablet with stylus	$295.00

We have The VIVA GOLD CARDS in stock ARTIST ONE compatability at a fraction of the price. Call for details.

ARTIST ONE is a registered trade mark of Control Systems.

The Computer Store Chain in Massachusetts went bust! We purchased their inventory of unused IBM equipment and supplies. We are able to pass along to you savings of up to 75%. Many items aren't listed. Call or write to us for your special requests. If you enjoy using, trading, or researching the hardware of science and technology, subscribe to our new magazine "The Science and Technology Collector". Get all the details in our latest 40 page issue. Send $3.00 for a sample copy. Your satisfaction is guaranteed! F & W Communications 241 E. Main Street Westborough, MA PO Box 509, 01581. Tel. (617) 366 0918

Degas Elite

continued from page 113

logically organized, more intuitive to use, and features a smoother, surer mouse interface than its predecessor.

Degas Elite allows you to draw in any graphics mode. You can draw freehand, or make use of tools that let you draw perfect lines, rectangles, circles, ellipses, polygons, boxes, discs, and rays. In addition to the airbrush offered in the original version, a "Stipple" selection has been added to Degas Elite, which is like an airbrush with a custom tip, which you can design. Another new feature, "Smear," allows you to semi-randomly mix-up the pixels within an area the size of your selected brush. It is handy for smoothing boundaries or smudging sharp contrasts into fuzzier ones.

"Mirror" allows you to draw symmetrically across up to six axes, while "Outline" draws an outline around whatever shape you specify. "Slowdraw" makes the mouse into a very precise instrument, while "Snap" allows you to align shapes to a grid. "Shadow" lets you offset another color from your main color in any of eight directions, with an offset of 1 to 16 pixels.

The "Fill" function of Degas Elite has been substantially refitted. In addition to the single-color fill patterns of the original program, pleasing and sophisticated multicolor fills are now available (in lo-res color mode). In addition, you can design your own. The fills are fast, and repeatable—you can always fill an area, even if it has been solid-filled before, as long as your fill pattern is of a contrasting color.

The palette and text functions have also been improved. Now colors can be moved from one paintbox to another, and "tweaking" between shades of a single color has been made much easier. Palettes alone are now loadable/savable, leaving the graphics screen intact. Conversely, images can be loaded, leaving the palette intact. Text fonts can be loaded as well, and text can be underlined, thickened, lightened, skewed (italicized), or outlined. Shadowed letters are possible, and can be used to dramatic 3-D effect.

Most impressive, however, are the new additions that now move Degas Elite from the realms of the "very good" to that of the "superlative."

"Magnify" expands an area of your drawing to about a third of the full-screen size so you can work on it in more detail. By pressing a function key, you can move from 3x to 12x magnification. At 12x magnification, each pixel is nearly the size of a dime. In the magnification mode, the top of the screen displays a palette, scrolling arrows, and a toggle for the level of zoom you would like. The left side of the screen shows the "real" size of your graphics, while the right side displays the magnified version.

You can draw in the magnify mode just as you would upon the regular drawing screen. Magnify gives you total control over every individual pixel on the screen of the Atari ST.

The concept of "Blocks" has been considerably expanded in Degas Elite, and offers a whole new range of capabilities. Pressing the [ESCape] key puts a crosshair on the screen, which can then be used to define a block. In this way graphics can be moved and copied. The "Stretch" command allows you to scale a block up and down vertically and horizontally. "Rotate" allows you to "turn" a shape to the angle you specify. Horizontal and vertical "Skewing" allows you to offset either the top and bottom or sides of a block. "Distort" lets you move each corner of a block to a different position on the screen and distort the image quite radically. You can even overlap sections of the block itself and "fold" the image upon itself.

In addition, individual blocks can be saved to and loaded from disk.

One of the most powerful additions to Degas Elite is the capability to provide color-cycling animation to graphics screens. "Animation" permits you to cycle a selection of colors in your palette automatically. If your imagery and palette are carefully orchestrated, the effect can be extremely effective. You can define up to four sets of animation palettes, and set individual speeds and "directions" for each.

"Cycle" is a drawing tool which uses the animation palette to define its colors. When you draw with a "Cycle" brush, the brushstroke will automatically cycle through the range of colors. This feature was once available only on graphics workstations costing thousands of dollars—and now literally shines on your Atari ST.

Perhaps most important, Degas Elite now supports multiple drawing or work-screens, depending on your memory configuration. A 520 ST will support four simultaneous screens, a 1040 will support up to eight. To select a screen to work in, you need only to click on a numbered box on the bottom right of the main menu.

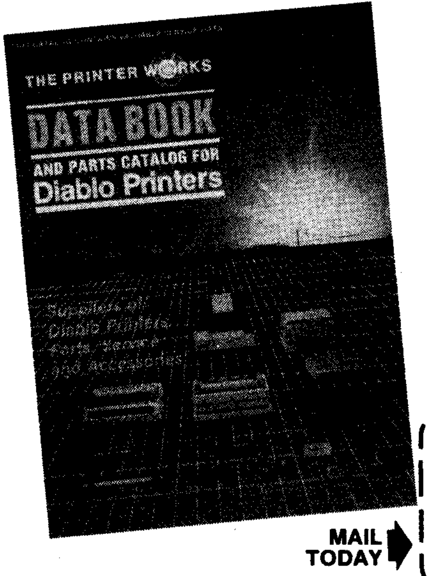
Multiple screens give you the capability to work up a library of graphics "primitives" for use in creating more complex drawings. You can load a number of screens simultaneously and move graphics between them. The only limitation is that all workscreens operate under the palette of screen 1.

The disk options from Degas Elite are also greatly improved. "Set Drive" allows you to set the default drive (not path) from which to load and save files. You can load pictures from NEOChrome and Koala Pad formats, as well as Amiga .IFF format pictures (using Electronic Arts' file format). You can also move images within ST resolution modes—for example, monochrome to lo-res color or hi-res color, or vice versa. When you load a picture saved in a different resolution from the one you're using, the picture is translated line by line, and Degas Elite processes and adjusts the image to achieve the best possible rendition of color densities.

Degas Elite also features an automatic compression routine which reduces the size of a standard picture file from roughly 34K to anywhere between 10 and 24K. There are several reasons to use or not to use compression. Some word processors now allow you to include graphic files such as Degas pictures in them. They usually require an uncompressed file. Translation programs which allow you to turn Degas pictures into other format files require uncompressed pictures to translate properly.

On the other hand, you can fit many more compressed pictures on a single disk, and Degas can read them without modification. If you are putting together a slide show, for example, the compression feature of Degas can increase the scope of your presentation quite a bit. (An improved slide

continued on page 248



FREE! THE MOST COMPLETE REFERENCE GUIDE ON DIABLO PRINTERS.

Not just a catalog, but a valuable source of reference data for those who own or service Diablo printers of any kind. *This manual offers:*

- ***Complete exploded diagrams of Diablo printers and key components***
- ***The most complete parts inventory outside of Diablo itself — including many hard to find parts***
- ***Plus new laser printing systems***

To receive your FREE copy of this comprehensive guide to parts and service of Diablo printers, just fill in the coupon below and send it to:

THE PRINTER WORKS

3481 ARDEN ROAD • HAYWARD, CA 94545

OR, CALL TOLL FREE

1/800/225-6116 (Inside California) **1/800/235-6116** (Outside California) **415/887-6116** (Local)

MAIL TODAY

Name ______________________

Company ______________________

Address ______________________

City ____________ State ________ Zip ________

NETWORK PC 1–800–638–1996 AFTER THE TONE 557648 (402)483–4965

Your Midwest Computer Dealer.....

PC OFFICE NETWORK
2.5 MEGABITS/SEC. $199.00/BOARD

EGA PACKAGE $699.00
14" EGA/CGA MONITOR W/SWIVEL & EGA/CGA/MGA GRAPHICS CARD

QMS KISS LASER PRINTER $1699.00
THIS AD WAS PRODUCED BY THE QMS KISS PRINTER.

XT CLONE CORNER

8 MHZ – 640K TURBO MOTHERBOARD	$ 99.00
10 MHZ – 640K TURBO MOTHERBOARD	$114.00
CGA COLOR CARD	$ 51.00
MONOGRAPHICS CARD	$ 59.00
FLOPPY CONTROLLER	$ 24.00
MULTI I/O CARD FLPY,SP,PP,CLK	$ 78.00
2–PORT SERIAL CARD	$ 39.00

1 year warranty all products.

AT CLONE CORNER

6–10 MHZ 1 MEG MOTHERBOARD	$399.99
FLOPPY CONTROLLER	$ 79.00
AT MULTIFUNCTION 2 MEG, 0K	$129.00
4–PORT SERIAL CARD	$ 89.00
EGA GRAPHICS ADAPTER	$199.00
MS–DOS 3.1	$ 55.00
THOMPSON RGB 580x300, .38 dot pitch	$269.00

PRICES SUBJECT TO CHANGE WITHOUT NOTICE. VISA, MASTERCARD & COD PRICES PLUS SHIPPING AND HANDLING

Graphics Editor

continued from page 244

replace or "cover up" whatever is behind the new shape, or it can be transparent, allowing the background to show through the newly overlayed shape. XOR (negative image) and reverse transparent (opposite of transparent) offer more control over the geometric shapes.

Placing these shapes on the screen is simple. Clicking the left mouse button places the shape on the screen. If you hold the left mouse button down while dragging the arrow accross the screen a series of the shapes are deposited.

No graphic page is complete without text, and this is where Typesetter ST and Rubber Stamp ST shift to high gear. Text mode is entered by depressing "T" while at the Graphic Card. This toggles you to the text mode and displays the Text Characteristics Card. The space bar removes the menu from the screen and the [F1] key will call it up again.

From the Text Characteristics Card (see Figure 1) you can select from as many as 23 different fonts or one that you have created with the font editor in DEGAS. Most of the options are straightforward so I won't go into them in much detail, but I really enjoyed the rotation command, allowing me to specify the rotation or orientation of the text to be inserted on the screen.

Rotations of anything other than 0 degree causes the cursor to disappear from the screen making text entry difficult. As long as you position the cursor first, before changing the direction, text entry is trouble-free.

Typesetter ST allows additional sub menus. Loaded font [F2], used to select graphic font qualities. Font height, width, color and inverse or overlay modes are supported. Large grid [F4], and fine grid [F5] are used to position the cursor to a relative pixel position.

Left margin [F7] and right margin [F8] allow you to adjust the margins to suit your taste. Move to a new cell [F9], selects the appropriate cell for the text mode.

Disk I/O functions [F10] supports loading of full Typesetter ST pages, standard 32K screens (such as NEOchrome or DEGAS pictures), or 8K icons or sketch pads created with Rubber Stamp ST and external fonts. Saving your files to disk is also straightforward, but be sure you have plenty of room on your disk, as a full graphic page takes up about 80K of space!

Rubber Stamp ST also supports an even larger array of sub-menus in text mode than Typesetter ST. [F1] allows selection from one of the 23 available fonts. Text rotation [F2] is used to rotate the text on the screen in any 90 degree orientation. As in Typesetter ST the cursor is only visible while the rotation is 0 degrees.

Thickness toggle [F3], sets the text to a larger thickness while screen move [F4] flips or overlays sections of the screen vertically or horizontally. Screen rotation [F5] is similar but, will rotate any section of the screen in 90 degree increments.

[F6] displays a fine grid on the screen to help position the cursor. Again, like Typesetter ST, after positioning the cursor the grid needs to be turned off before any other action can take place. Margins are easily adjusted by [F7] for the left and [F8] for the right.

Rubber Stamp ST supports a wide array of printer selections [F9]. Selectable outputs are Full screen dumps, 15/16 x 3-1/2 inch label, 2 x 2-3/4 inch Rolodex card, 3 x 5 inch index card, 4 x 6 inch index card and free form.

Disk I/O function [F10], allow you to load or save the screen as either a sketch pad area or an entire screen. When loading or saving a sketch pad, the quadrant must be specified. External fonts can be loaded and so can screens from a converted NEOchrome or DEGAS picture.

It should be apparent that Typesetter ST as well as Rubber Stamp ST support many commands. To really get an appreciation for the programs you have to use them.

Four utility programs are provided in Typesetter ST and Rubber Stamp ST. With the utilities provided, DEGAS (Batteries Included), NEOchrome (Atari) and Snapshot (Michtron) files can be converted to a 32K Typesetter or Rubber Stamp ST screen format for use with Typesetter ST or Rubber Stamp ST.

LOWMED.PRG converts a low resolution screen to a medium resolution Typesetter ST or Rubber Stamp ST file. Xlent recommends the use of a .MMM file extender on converted files. LOWVER.PRG converts low resolution file to a high resolution file using an extender of .HHH. MEDVERT.PRG converts medium resolution files to high resolution files and also carries the .HHH extender. The last utility is an Epson JX-80 printer driver.

Typesetter ST and Rubber Stamp ST are fun to use and serve a useful purpose. Each program seems to complement the other and as a pair they give you a nice graphic editor for full graphic page printouts at a reasonable price.

This is not to say that the software is without problems. Numerous small bugs are present. Typesetter ST seemed to have more problems than did Rubber Stamp ST but, being the first release of the program they're no buggier than expected. Rubber Stamp ST seemed much cleaner and it is apparent from the additional features available, the product is maturing quite nicely.

Both manuals for the programs were written by Arthur Leyenberger and seemed to cover most of the material well. Even with the complexity and power of the programs, by following the manual a step at a time I was able to master them both.

Typesetter ST and Rubber Stamp ST are distributed by XLent Software, P.O. Box 5228, Springfield, VA 22158; (783) 644-8881. ●

Degas Elite

continued from page 247

show program is included on the Degas Elite disk, along with a set of printer drivers, and a font editor.)

All commands except those on the pull-down menu bar can be activated from the keyboard. This is an excellent feature, and one we have come to expect from Batteries Included. It ensures that as the user advances, the point-and-click interface that initially made the product easy-to-learn does not itself later become a disadvantage. The user can move from point-and-click to selected keyboard commands at his own rate.

The documentation accompanying the package is clear, well-written, and well-organized. If I was to pick any nits, it would be to point out that its index can be a bit maddening—it points you to every occurrence of the word you seek, with no hint of where the "meat" will be found. One example: "Disk 1, 3, 4, 9, 14, 43-46, 51, 53-56, 59-62, 64, 66, 67, 69, 81." This makes for an impressive-looking index, but doesn't really help the user very much. (It is a frequent problem, now that indices are almost always compiled by computer.) The documentation is also perfect-bound, which means you will have to break its spine to get it to lay flat and open. Pages are coming loose from mine now, so please treat yours with more respect.

But it is the software that counts, and Degas Elite is probably the best piece of software to have appeared for the Atari ST to date. If you have Degas and use it regularly, you will want to upgrade. If you have held out on a paint program for the Atari, let it be known that time has come.

Tom Hudson, the author of Degas and Degas Elite, has scored once again—this time with a program that leaves absolutely nothing to be desired. Now if only he would get to work on an animation program of the same caliber...

Degas Elite retails for $79.95. It is *not* copy-protected. For more information, contact Batteries Included, 30 Mural St., Richmond Hill, Ontario, Canada LB4 1B5. (416) 881-9941. ●

Sinclair Survival Column

continued from page 125

may slice portions of it to suit your needs, such as placing some of this information in a separately created window onscreen which constantly displays the current information. (Next month I will show you how to accomplish this.) Using the function DATE (no $) will return the date as a floating point number which you may use to store dates and times in a more compact form.

To get the day of the week you would use the function DAY$ which will use the date information to compute and present the current day of the week.

When using ARCHIVE (database) you may want to create a form such as an invoice, purchase order, or any other form or report on which you require the time and date of its printing. Although ARCHIVE uses its own set of commands for this, you still must set the QL's clock upon powerup in this fashion before you load ACHIVE.

A final command relating to the clock functions is ADATE. ADATE is used to adjust the clock without going through the SDATE function. ADATE followed by a number of seconds (may be positive to set the clock ahead, or negative to set the clock back) will make fine adjustments. For example—ADATE 3600 will advance the clock by one hour while ADATE-60 will move the clock back one minute.

This is just our first look at the power of SuperBASIC which will continue on an ongoing basis. If you have a short program which can be used as a demonstration of some feature of the QL, QDOS, SuperBASIC or any other aspect of the computer's operation, send it to me in care of *Computer Shopper* and I will share the best ones. ●

X'MAS SPECIAL SALE ON ALL IBM PC/XT/AT CLONE SYSTEMS!!

$695! Monochrome System Price Includes:

- Full 640K RAM
- Standard & Turbo mode
- Two 360K floppy drives
- Monochrome high resolution monitor
- Monochrome graphics adaptor
- Serial, Parallel, Game ports w/clock
- Keyboard
- Industry Standard Pheonix ROM BIOS
- 135W Switching Power Supply
- 8 Expansion Slots
- 8087 MATH Co-processor slot
- 48 hour burn-in
- Runs all Major PC-DOS Software
- 1 Year Warranty

FREE DOS 3.1 AND A SET OF PUBLIC DOMAIN GAMES

Add only $200 to obtain the above system with RGB Color Card and RGB Hi/Res Color Monitor. Add only $399 to add a 20MB hard drive and $499 to add a 30MB hard drive.

Complete AT System with Monitor............................$1395 with 640K memory, 1.2MB & 360K Floppys, Monochrome Hi/Res Monitor with Adaptor & printer port, Keyboard. Add only $200 to obtain a Color System, $499 to add a 20MB hard drive and $599 to add a 30MB hard drive.

To Order Call
TOLL FREE 1-800-331-0488
1-305-296-6083 in Florida
SSL Inc.
103, Fitzpatrick St.
KeyWest Florida 33040

! Note: We gaurantee our prices to be the lowest and will beat any comparable advertised price by $10.00. These prices good while the quantities last, so rush and book your Christmas special order now.

SSL's Policy: We accept MasterCard and Visa, money order, cashier's check, personal check (allow 10 days to clear). We charge 5% for shipping and handling by UPS ground (faster shipping available at additional charge. Any item bought from SSL may be returned for a full refund within 30 days of date it was shipped if for any reason you are not 100% satisfied. All items under warranty must be returned after obtaining Return Authorization Number from us by telephone.

IBM is a registered trademark of IBM Corporation.

QUANTITY DISCOUNT AVAILABLE